REMARKS

Upon entry of this amendment, claims 1-4, 6-18, 20 and 21 will be pending. (Claims 5 and 19 have been previously cancelled without prejudice.) Claims 1, 12, 18, 20, and 21 are currently amended. These amendments introduce no new matter, and are supported in the specification at, for example, page 7, lines 5-13; page 8, line 23-28; page 9, lines 2-10; page 13, lines 1-12; and Fig. 4, references 13 (software engine) and 19 (software components or objects).

The independent claims have been amended to more clearly recite parsing of hierarchically-structured (specifically, XML) input streams in which objects mapped to tags are built into a partially-built tree representation of the initial, already-parsed portion of the input stream as soon as their associated tags are recognized during parsing, and in which methods defined by the objects built into the tree are invoked also during parsing. Importantly, the independent claims have also been amended to more clearly recite that the invoked methods have awareness of the position of their associated tags in the partially-built tree representation, but have no awareness of a complete tree representation of the entire input stream.

The invoked methods have such awareness of the partially-built tree representation for the following reasons. First, methods have an awareness of the tree in which their defining objects are built, because they are invoked only after their defining objects have been built into the tree. Second, because the objects are built into the tree as soon as their associated tags are recognized in the input stream being processed, this tree can be a representation of only the already parsed portion of the input stream. It cannot represent the entire input, and the invoked methods can therefore have no awareness of a complete tree representation of the entire input stream.

In the Office action, claims 1-5, 7, 8, 11, 12-17, and 18-21 were rejected under 35 U.S.C. § 102(e) as being anticipated by US patent no. 6,675,354 by Claussen et al. ("Claussen"). This rejection is traversed because Claussen does not disclose each and every element recited in the amended claims.

Claussen does not disclose or teach the claimed methods and systems during which parsing, object building, and method invocation occur together. Instead, Claussen discloses a sequential two-step processing of XML streams. In the first step, an input XML document is parsed and transformed into a hierarchical DOM (tree structured) representation. See Claussen at, for example, col. 5, lines 25-30; and col. 5, line 48 to col. 6, line 6; and Fig. 2, references 204 and 206 in Fig. 2. Then, in the second, later, and entirely separate step, the DOM representation is traversed to produce and invoke JAVA software components. See Claussen at, for example, col. 6, lines 8-20; and Fig. 2, references 206-214.

Consequently, in contrast to all the independent claims, Claussen's methods do not have an awareness of a tree representation of only an initial, already-processed portion of the input stream, because they are not invoked during the parsing process when the tree being built represents only the initial, already-parsed portion of the input stream. Instead, Claussen's methods are only invoked during a second pass, when the tree representation of the input stream has been fully built, so that they, at most, have awareness only of a complete tree representation of the entire input stream.

In summary, because Claussen does not disclose each and every element recited in the independent claims, *inter alia*, that the invoked methods have awareness of a tree representation of only an initial, already-processed portion of the input stream, these claims are not anticipated by Claussen. Further, the dependent claims are also not anticipated, because they inherit the patentable limitations of their parent independent claims.

The Office action also rejects claims 6, 9, and 10 are being unpatentable under 35 U.S.C. § 103(a) over Claussen in view of US patent no. 6,125,391 by Meltzer et al. ("Meltzer") or of US patent no. 6,434,529 by Walker et al. ("Walker"). The rejection is traversed because neither Meltzer nor Meltzer supplement Claussen's deficiencies.

In Meltzer, any methods that might be invoked concurrently with parsing an XML input stream are defined by objects that are not built into, and do not have any relationship whatsoever with, a tree representation of an initial, already-parser portion of the input stream (or indeed even a tree representation of the entire input stream). Thus, in contrast to the independent claims, any

methods invoked by Meltzer cannot have any awareness of the position of their associated tags in a tree representation an initial, already-parsed portion of the input stream.

In more detail, Meltzer not only does not disclose building objects into a tree representation of an initial, already-parser portion of the input stream, but for the following reasons, Meltzer's methods and systems are such as to prevent building any such relationships. With reference to Fig. 5, parser 500 generates "XML events" and sends generated "events" to a router for further relay to "listening" programs at various network locations. See Meltzer at, for example, col. 23, lines 64-66; and col. 26, line 63 to col. 27, line 2. One listener, listener 506, builds a tree representation of the XML input, while other separate listeners, listeners, 503, 504, and 505, perform event-associated application processing. See Meltzer at, for example, col. 24, line 59 to col. 25, line 3; and col. 28, lines 8-9 and 15-20.

However, these listeners are entirely independent and asynchronous of each other. Fig. 5 especially illustrates that listeners 503, 504, 505, and 506 are independent parallel processes without any mutual influence or awareness. Also see Meltzer at, for example, col. 25, lines 58-64. Hence, because tree builder 506 is independent and asynchronous of, and without any awareness of or influence from, parser 500 or other listeners 503, 504, and 505, it is not possible for tree builder 506 to build into the tree objects associated parser 500 or with other listeners 503, 504, and 505. It simply has no knowledge of or access to any objects to build into a tree. Thus, Meltzer's system cannot invoke methods defined by objects associated with parser-recognized tags with an awareness of the position of their associated tags in a tree representation of the initial, already-processed portion of the input stream.

Finally, Walker, being exclusively concerned with speech recognition grammars, does not disclose or teach any hint of these limitations of the independent claims, all of which are exclusively concerned with markup languages.

In summary, because the combination of Claussen, Walker, and Meltzer does not, *inter alia*, disclose or teach every element recited in the independent claims, this combination does not establish the obviousness of these claims nor of their dependent claims.

In view of the foregoing, Applicants respectfully submit that all the Examiner's objections and rejections have been addressed and that all of the claims in the present application are allowable. Accordingly, Applicants respectfully request that the claims be reconsidered and passed to allowance.

DATE: 7/29 ,2004

Paul D. Greeley, Esq.
Attorney for Applicant(s)
Registration No. 31,019
Ohlandt, Greeley, Ruggiero & Perle, L.L.P.
One Landmark Square, 10th Floor
Stamford, CT 06901-2682
Telephone: (203) 327-4500
Telefax: (203) 327-6401